JÜLICH
FORSCHUNGSZENTRUM

Report on Project: Advanced System Monitoring for the Parallel Tools Platform (PTP)

September, 2014 | Wolfgang Frings and Carsten Karbach

Project progress

- ✓ Server caching
- Customized display layouts
- Optimized job localization
- Usage history and scheduling prediction

Content

1. LLview/PTP overview
2. Server caching
3. Display layouts
4. Optimized job localization
5. Embed LLview diagrams

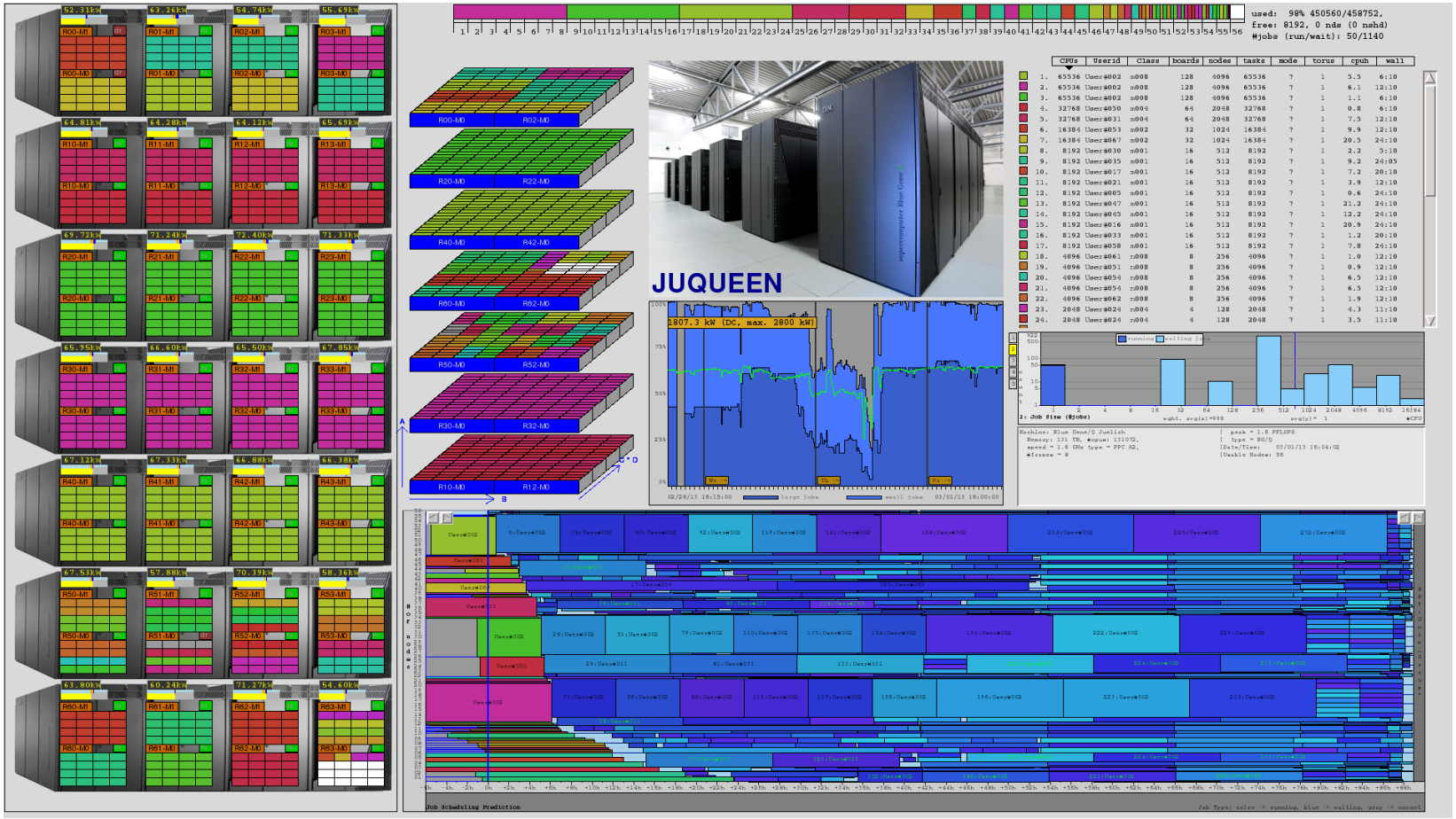
Part I: LLview/PTP overview

September, 2014 | Wolfgang Frings and Carsten Karbach

JÜLICH FORSCHUNGSZENTRUM

LLview

→ Visualizes supercomputer status on a single screen



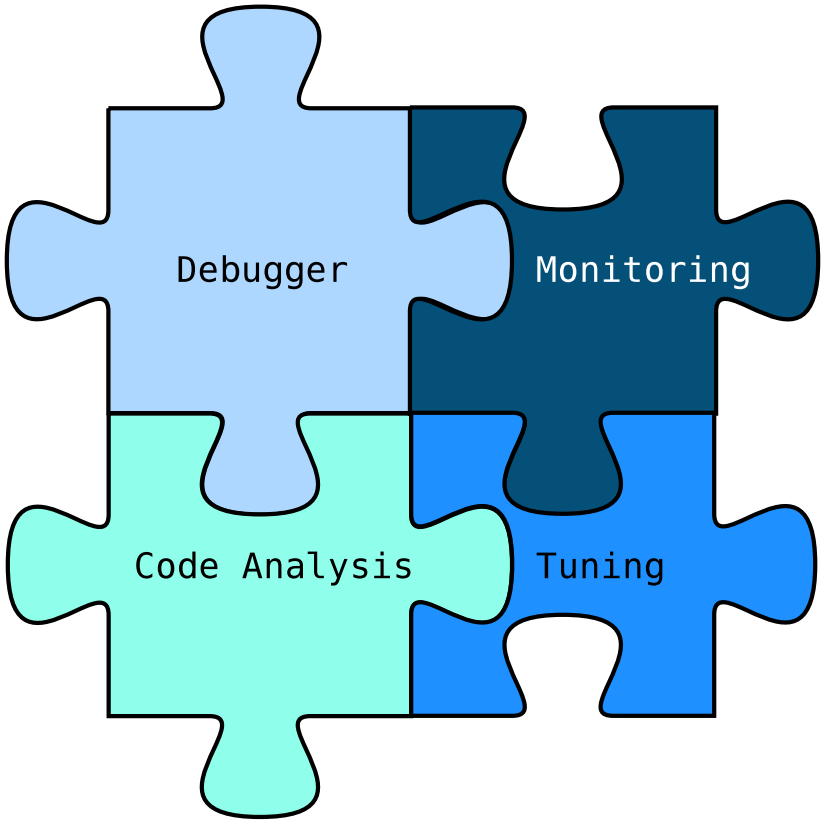
Source: Screenshot LLview for JUQUEEN

JÜLICH FORSCHUNGSZENTRUM

PTP – Parallel Tools Platform

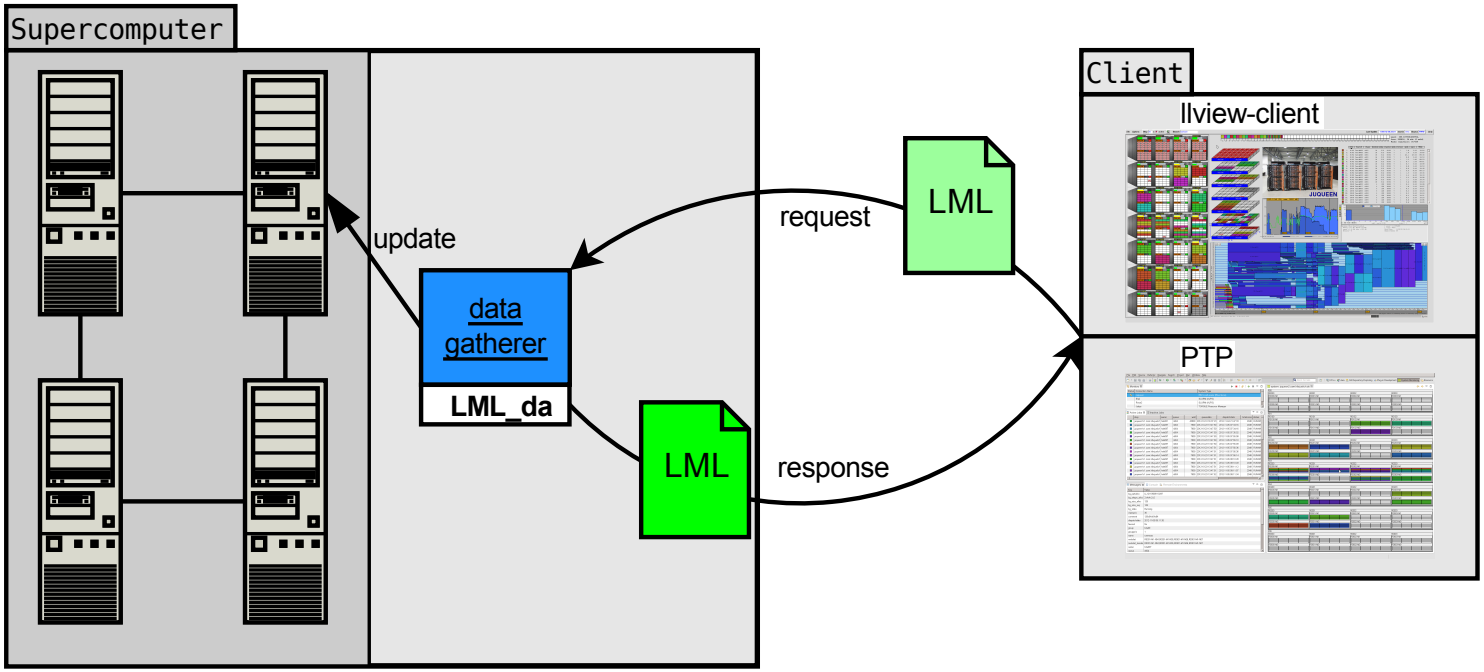
What is PTP?

- **IDE** for parallel application development
- Based on **Eclipse**
- **Open-source** project
- Developers: IBM, U.Oregon, UTK, Heidelberg University, NCSA, UIUC, JSC, ...



JÜLICH FORSCHUNGSZENTRUM

Monitoring Architecture



Part II: Server caching

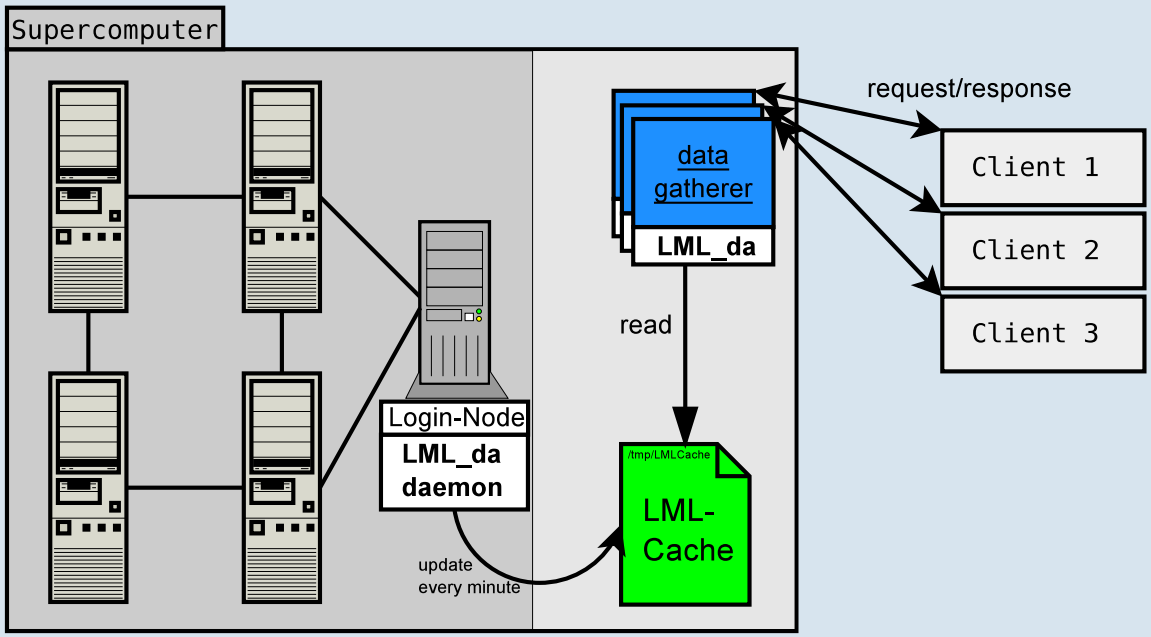
September, 2014 | Wolfgang Frings and Carsten Karbach

JÜLICH FORSCHUNGSZENTRUM

Architecture

- **multiple users** on the same target system
- **cache** LML file in public directory (e.g. /tmp), use LML cache as data source
- default: each client triggers independent status data update
- caching: daemon retrieves status data, clients use cached data

Cache workflow



JÜLICH FORSCHUNGSZENTRUM

Implementation

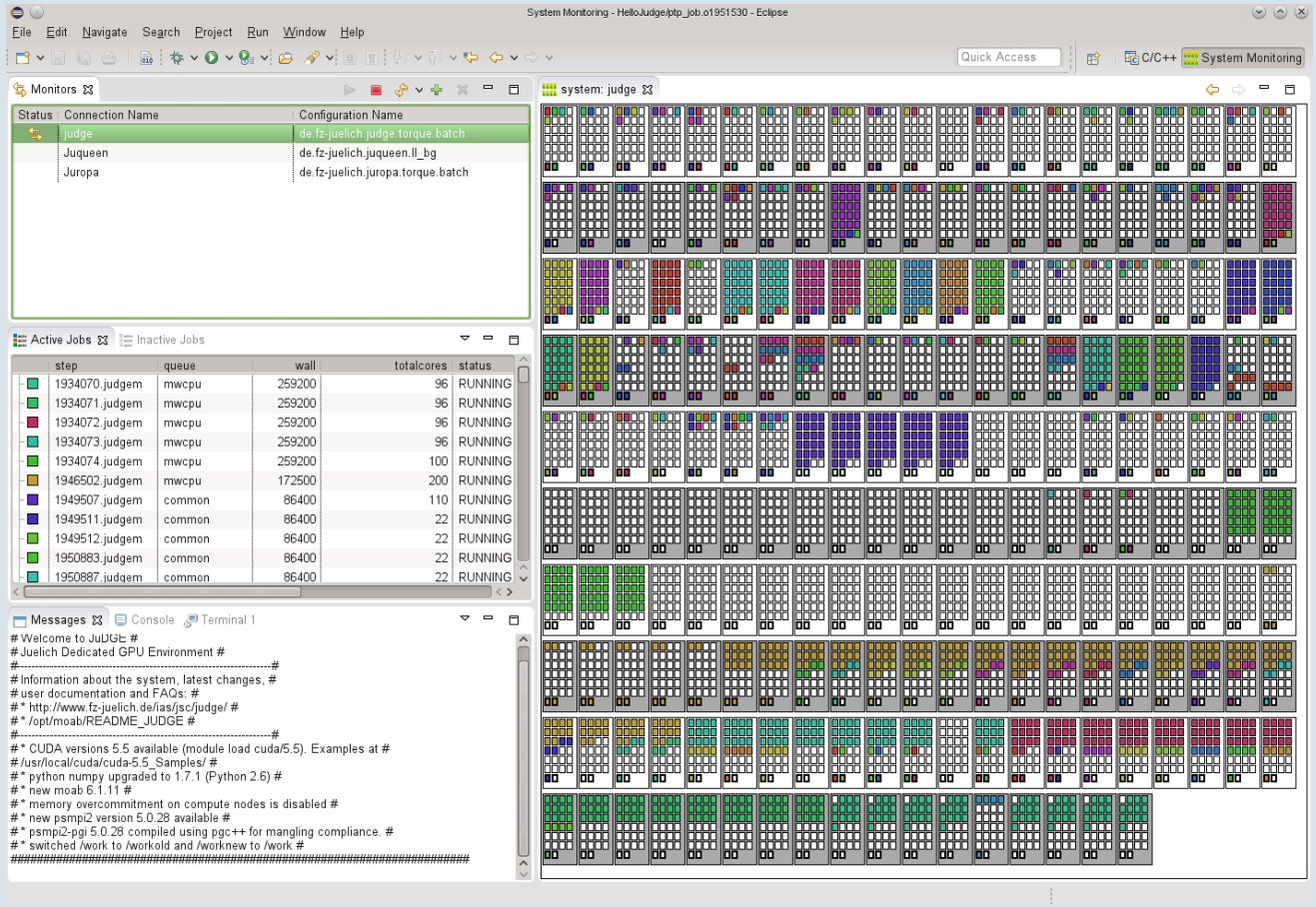
- LML caching is included in latest Eclipse release Luna (July 14) from `http://www.eclipse.org/downloads/`
- Implementation is documented on bug 427386

Install daemon, once for each system

1. Create monitoring connection and start it
2. SSH to target system, switch to `.eclipsesettings/util/install`
3. Call install script: `perl installCrontab.pl`
4. Run `util/install/LML_da_crontab.sh` every minute

Install step 1

Start monitoring connection



JÜLICH FORSCHUNGSZENTRUM

Install steps 2, 3, 4

Install daemon and run it

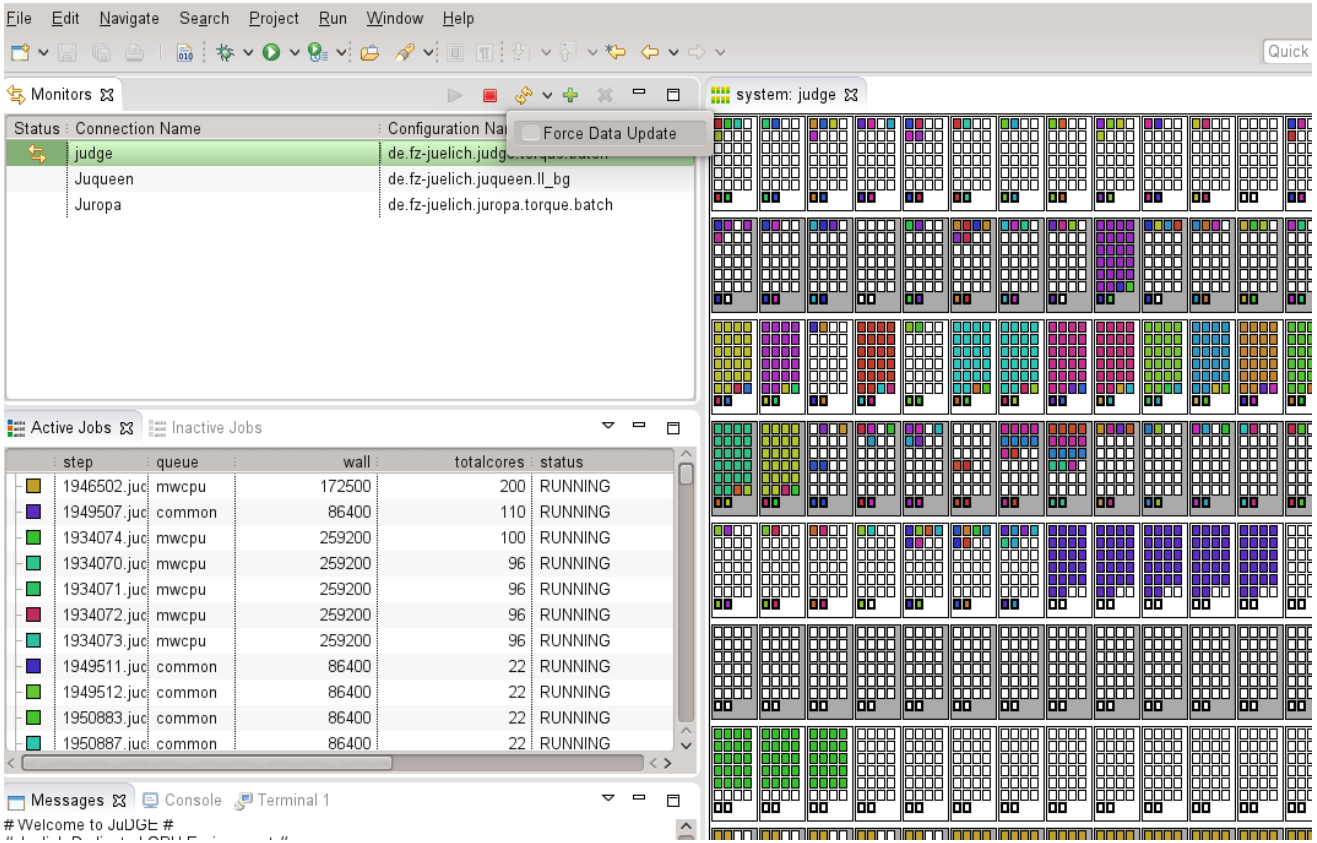
Problems Tasks Console Properties Call Graph Terminal 1

```
judge
karbach@judge:~> cd .eclipsesettings/util/install/
karbach@judge:~/.eclipsesettings/util/install> ll
total 96
-rw-r--r-- 1 karbach zam  1703 2014-03-31 17:30 config
-rw-r--r-- 1 karbach zam   107 2014-07-30 08:16 crontab.add
-rwx------ 1 karbach zam   112 2014-07-30 08:16 daemon_script.sh
-rw-r--r-- 1 karbach zam 10331 2014-03-31 17:30 installCrontab.pl
-rwx------ 1 karbach zam   263 2014-07-30 08:16 LML_da_crontab.sh
-rw-r--r-- 1 karbach zam  1491 2014-03-31 17:30 README
karbach@judge:~/.eclipsesettings/util/install> perl installCrontab.pl
Installation options: croninterval => "* * * * *" rms => "undef" installdir => "" cachedir => "/tmp/LMLCache_judge/" permdir => "./perm_cac
he_judge/"

1) Check the options configured in the ./config file
Detected RMS "TORQUE"

2) Generate the script called repeatedly, which generates the raw LML file
LML_DA directory is "/homeb/zam/karbach/.eclipsesettings/util/install/../../"
Generating script for crontab in the file "/homeb/zam/karbach/.eclipsesettings/util/install/LML_da_crontab.sh"
Making the crontab script executable

3) Create a file containing the cron job
Generating a file named "crontab.add" containing the cronjob, which has to be added to crontab.

4) Create a daemon script as alternative for crontab
Generating daemon script as alternative for crontab in "/homeb/zam/karbach/.eclipsesettings/util/install/daemon_script.sh"
Making the daemon script executable.

5) Explain, how to add the cronjob and how to use the daemon script
Please append the content of "/homeb/zam/karbach/.eclipsesettings/util/install/crontab.add" to your crontab configuration
Alternatively you can run the script "/homeb/zam/karbach/.eclipsesettings/util/install/LML_da_crontab.sh" repeatedly with any other mechani
sm.
E.g. the script "/homeb/zam/karbach/.eclipsesettings/util/install/daemon_script.sh" can be used for that purpose.
Make sure, that the status data in "/tmp/LMLCache_judge/" is continuously updated after you have finished the installation.
Note, that a crontab entry would be preferable as it is more reliable than the daemon script.
karbach@judge:~/.eclipsesettings/util/install> ./LML_da_crontab.sh
  LLVIEW Data Access Workflow Manager Driver 1.18, starting at (Wed Jul 30 08:24:11 CEST 2014)
  LLVIEW Data Access Workflow Manager Driver 1.18, ending at   (Wed Jul 30 08:24:11 CEST 2014)
karbach@judge:~/.eclipsesettings/util/install> ll /tmp/LMLCache
LMLCache/       LMLCache_judge/
karbach@judge:~/.eclipsesettings/util/install> ll /tmp/LMLCache_judge
total 472
-rw-r--r-- 1 karbach zam  15908 2014-07-30 08:24 colormap.db
-rw-r--r-- 1 karbach zam 460210 2014-07-30 08:24 LML_raw.xml
karbach@judge:~/.eclipsesettings/util/install>
```

JÜLICH FORSCHUNGSZENTRUM

Client usage

- Default update mode: active caching
- If no daemon installed ⇒ no caching
- Users can deactivate caching via *Force Data Update*



JÜLICH FORSCHUNGSZENTRUM

Advantages of caching

- Faster response time

System	Cache [s]	No Cache [s]
JUDGE	1.1	2.8
JUROPA	5.8	11.1
JUQUEEN	1.7	34.4

- Recording of history data is possible
- Enhancement with data not accessible to normal user
- Decreased load for the system

Part III: Display layouts

September, 2014 | Wolfgang Frings and Carsten Karbach

JÜLICH FORSCHUNGSZENTRUM

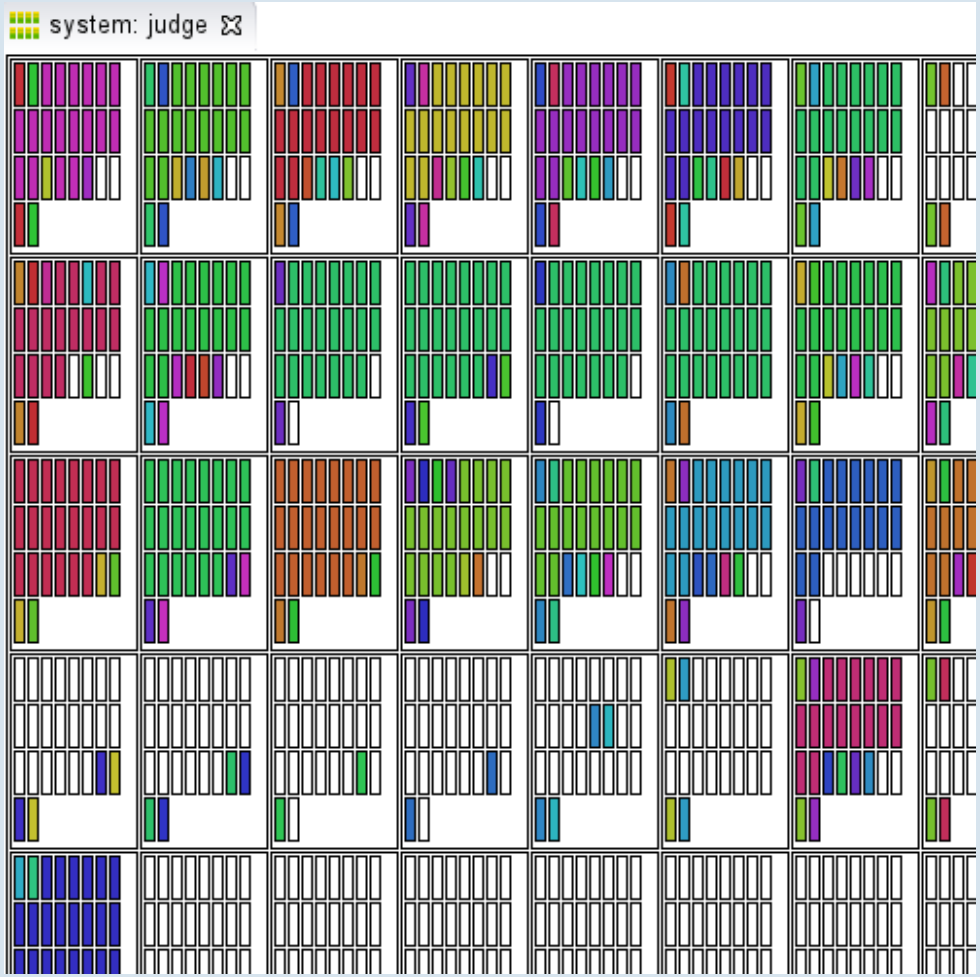
Customized LML layouts

- understand system architecture and hierarchy
- map topology into LML layout
- advantages: level of detail, automatic job filtering, display node names, improved performance
- workplan: tutorial on LML layouts, contact system administrators of partner XSEDE/PRACE sites, support writing of LML layouts, ask for feedback

JÜLICH FORSCHUNGSZENTRUM

Example JUDGE

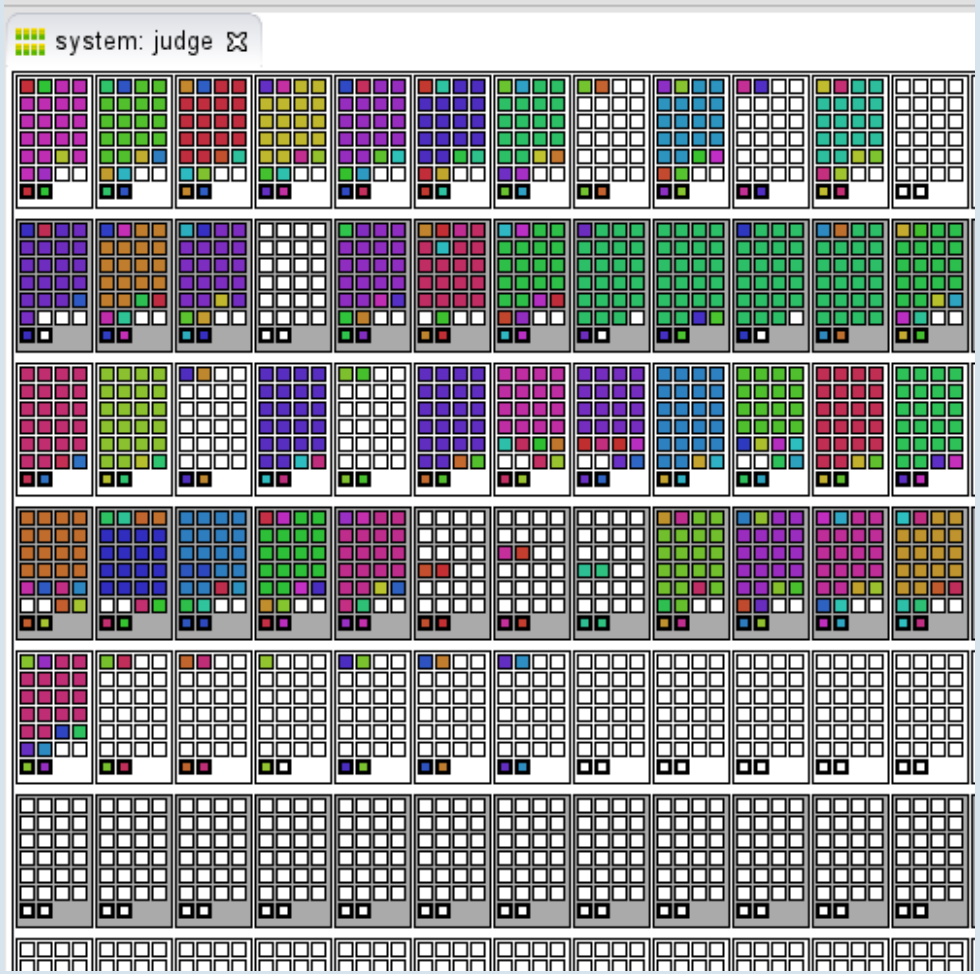
Default layout



JÜLICH FORSCHUNGSZENTRUM

Example JUDGE

Customized layout

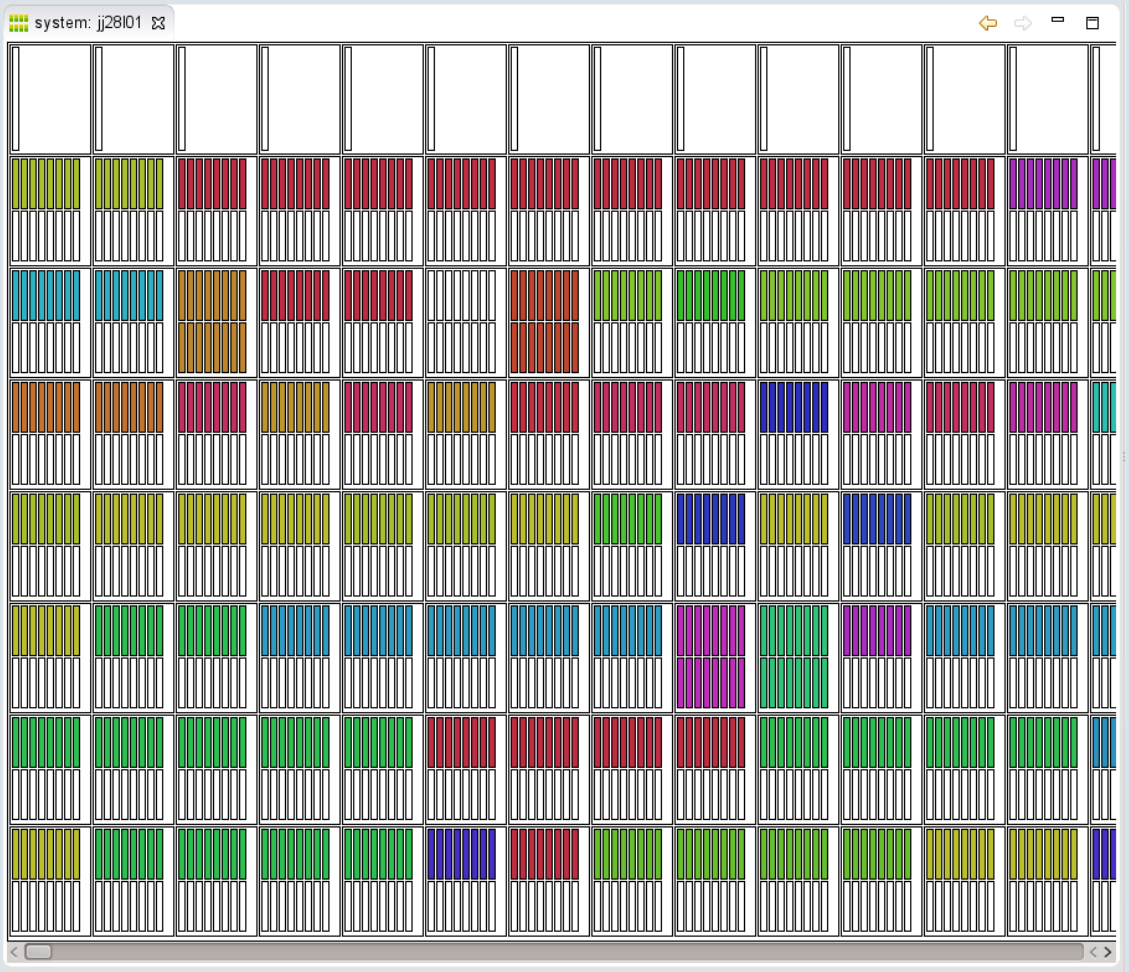


- alternating background colors
- improved grid for square display
- highlighted GPUs

JÜLICH FORSCHUNGSZENTRUM

Example JUROPA

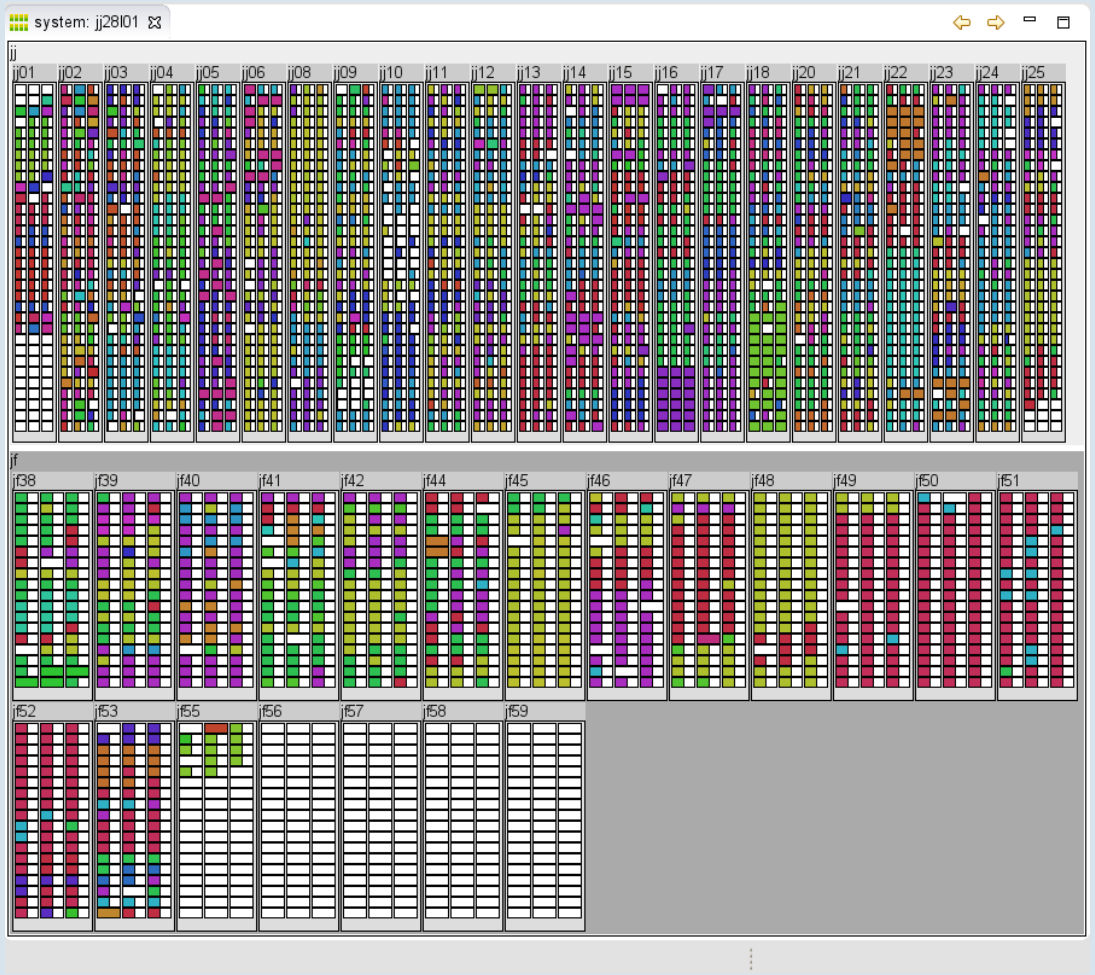
Default layout



JÜLICH FORSCHUNGSZENTRUM

Example JUROPA

Customized layout

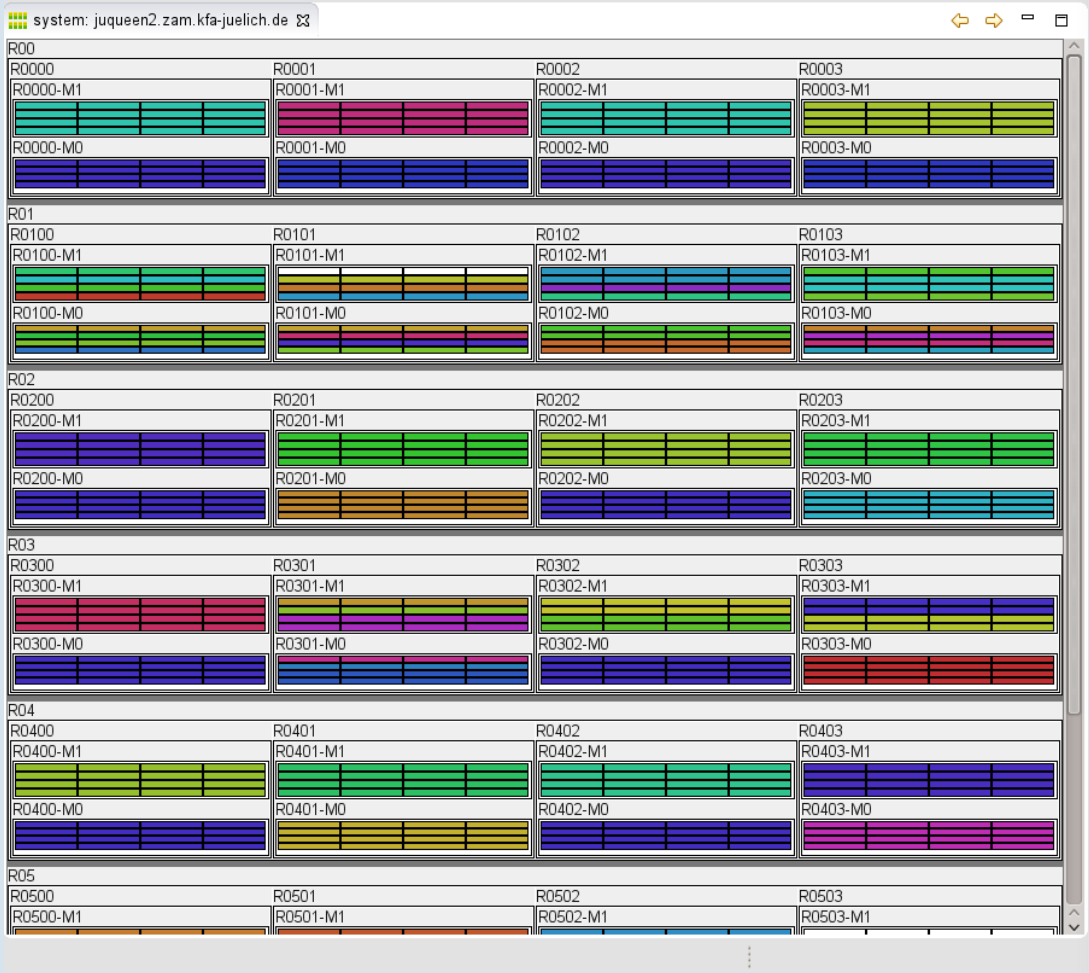


- four level hierarchy: partition, rack, node, core
- level of detail (only with layout)
- better overview

JÜLICH FORSCHUNGSZENTRUM

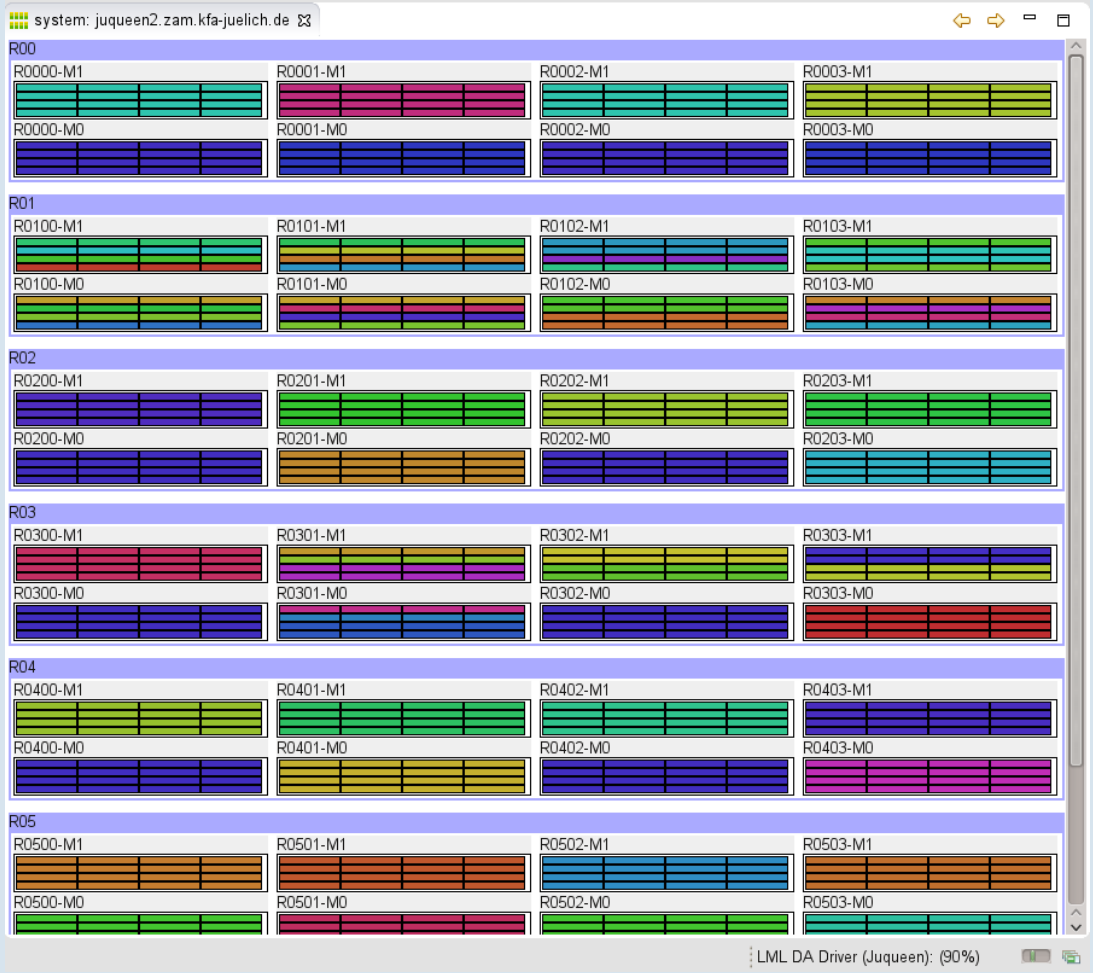
Example JUQUEEN

Default layout



Example JUQUEEN

Customized layout

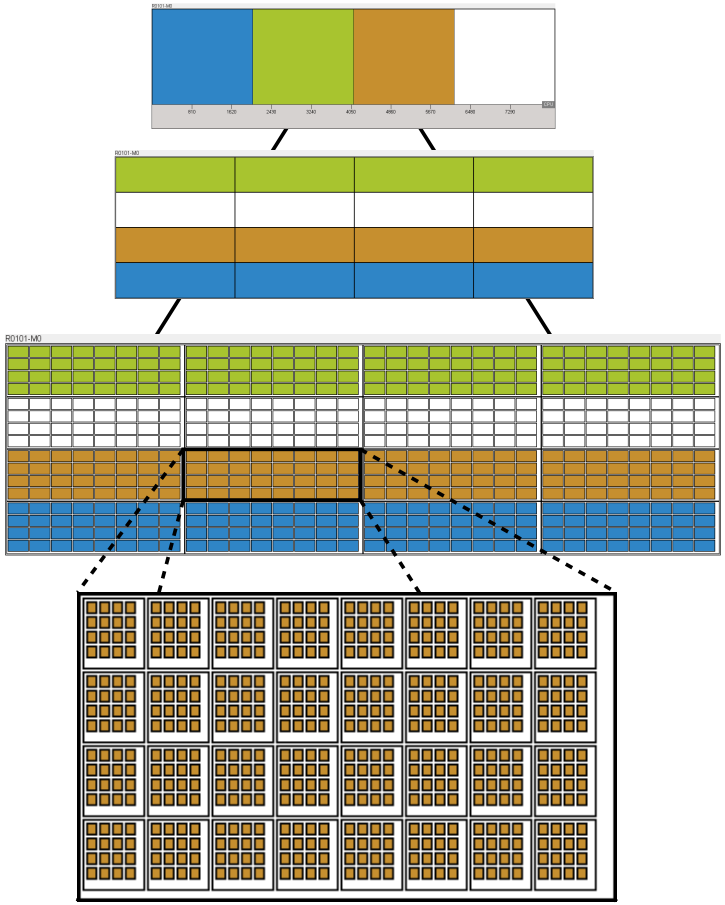


- colored rows
- do not display rack names

JÜLICH FORSCHUNGSZENTRUM

Scalability

1. Scalable **server scripts** LML_da
   - Select only required data
2. Scalable **data format** LML
   - Data compression
   - Redundancy avoidance
   - Uses system hierarchy
3. Scalable **visualization** PTP
   - Filter status data
   - Levels of detail
   - Zoom function



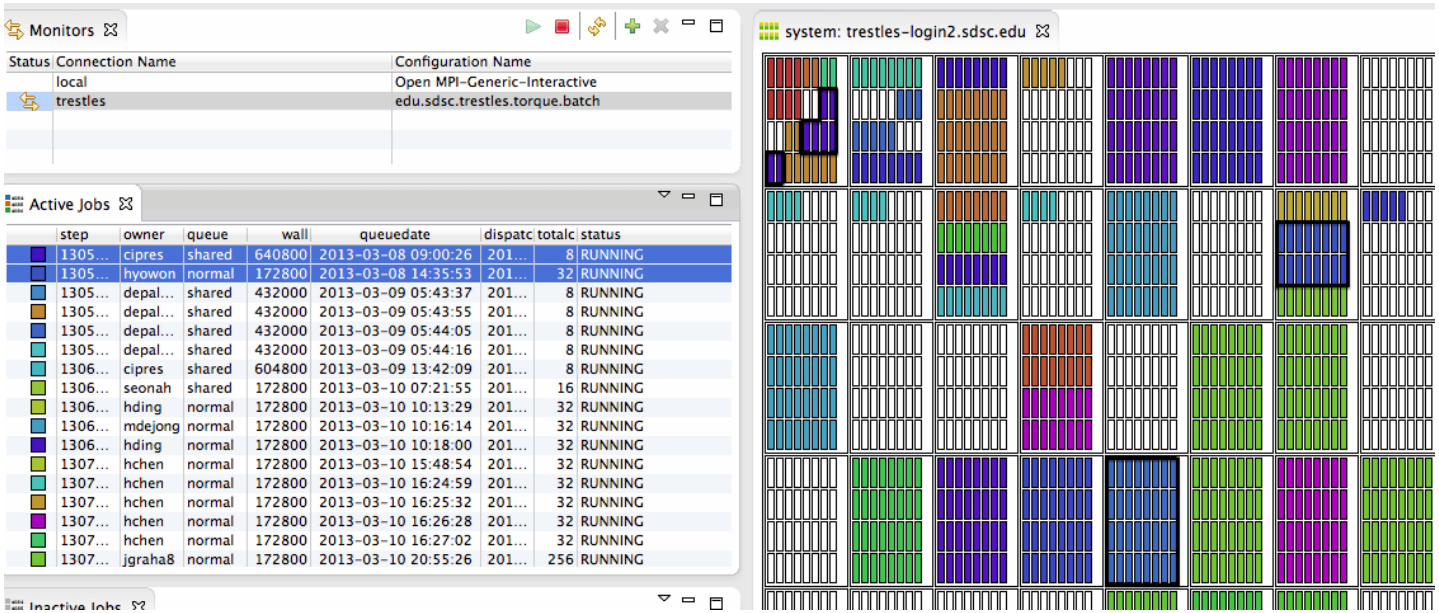
Part IV: Optimize job localization

September, 2014 | Wolfgang Frings and Carsten Karbach

JÜLICH FORSCHUNGSZENTRUM

Job selection

- raised in bug 403060
- allow selection of **multiple** jobs
- keep selected job selected until it is de-selected
- mark entire **connected area** of each job



Source: `https://bugs.eclipse.org/bugs/attachment.cgi?id=228316`

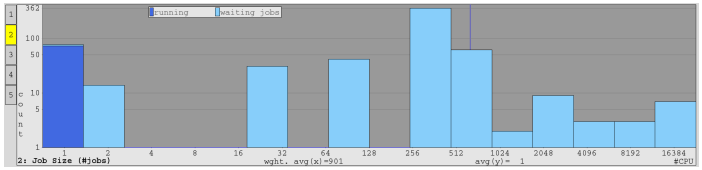
Part V: Embed LLview diagrams

September, 2014 | Wolfgang Frings and Carsten Karbach

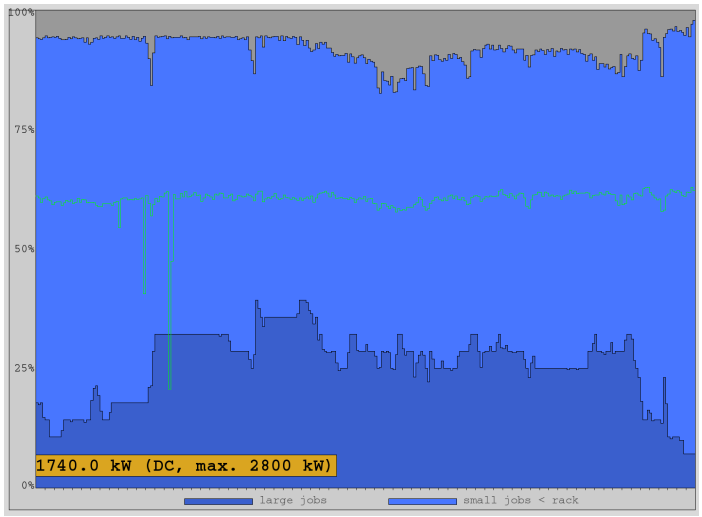
JÜLICH FORSCHUNGSZENTRUM

LLview diagrams

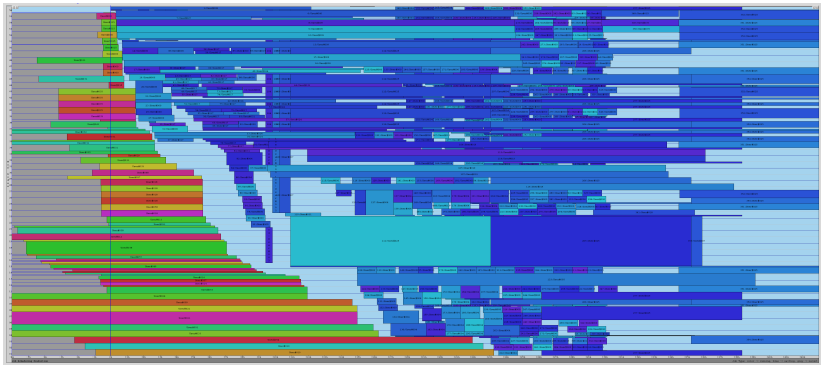
- histograms



- load history



- prediction

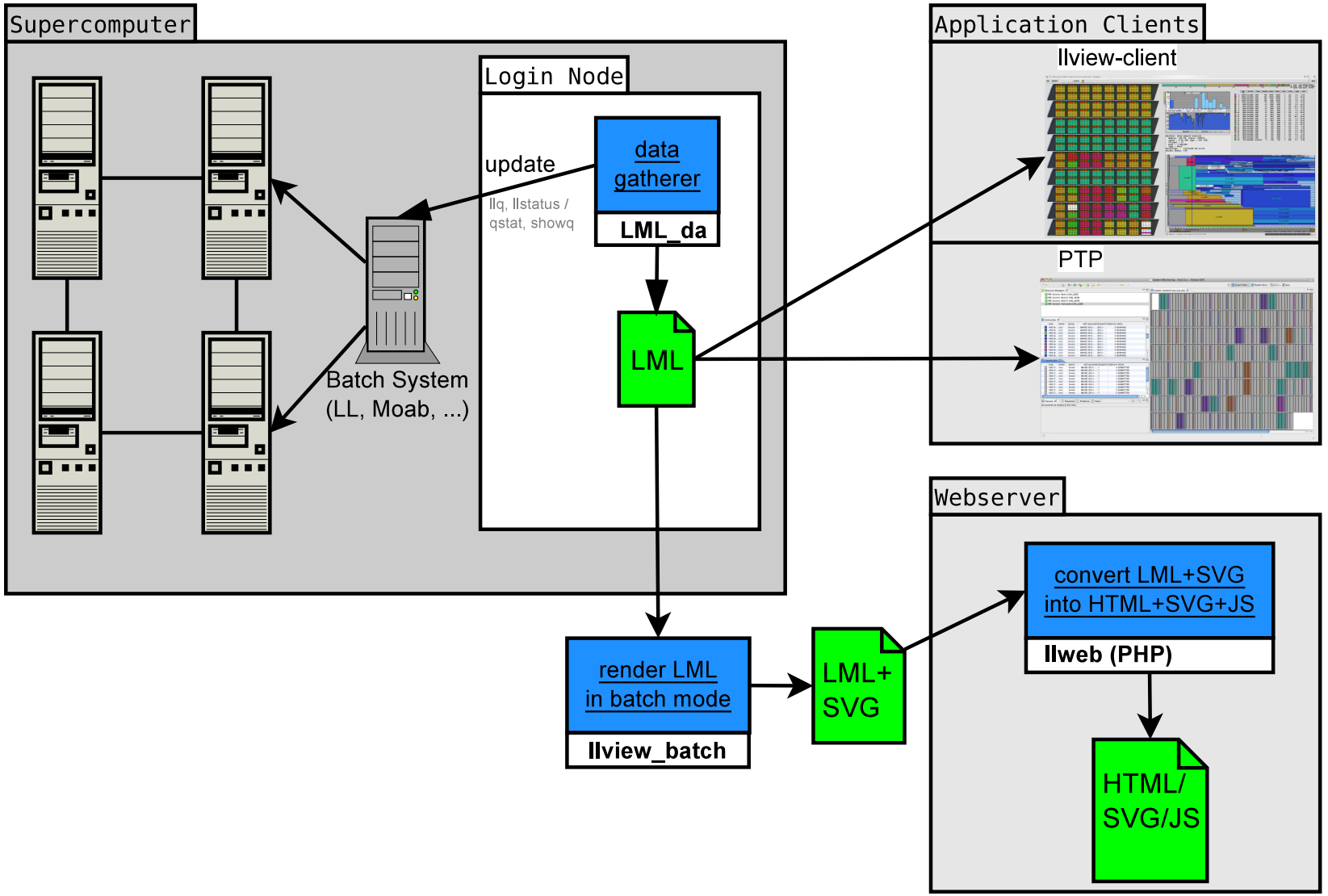


Challenges

- Re-implementation in Java?
- Double update for LLview and PTP

JÜLICH FORSCHUNGSZENTRUM

First implementation step



JÜLICH FORSCHUNGSZENTRUM

Evaluation

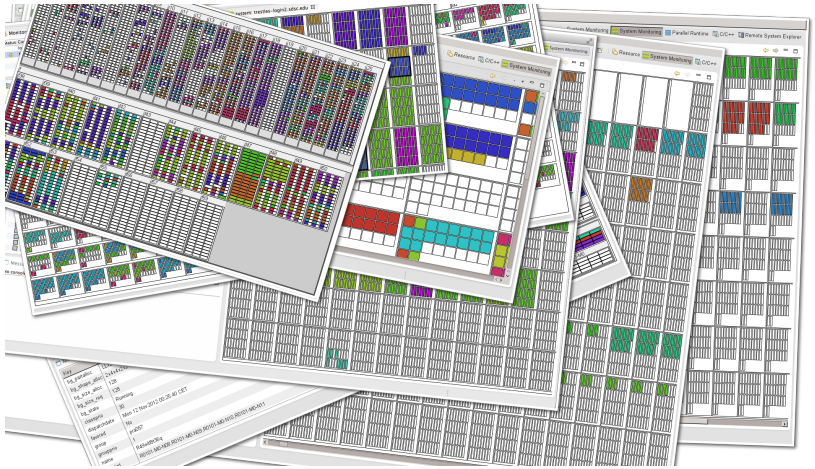
Advantages

- expecting little implementation effort
- single source of diagram implementation
- use of SVG graphics within PTP and web client
- SVG graphics can contain tags for interactive data analysis

Problems and risks

- adding LLview-client to PTP source, license
- analysis of SVG performance and support

Contact



- **E-mail**: c.karbach@fz-juelich.de, w.frings@fz-juelich.de
- **PTP** → `http://www.eclipse.org/ptp/`
- **LLview** → `http://www.fz-juelich.de/jsc/llview`